

**REMARKS**

Reconsideration and allowance of the subject application are respectfully requested. By this Amendment, Applicant has amended claim 4. Upon entry of this Amendment, claims 1-17 are all the claims pending in the application. In response to the Office Action, Applicant respectfully submits that the claims define patentable subject matter.

Claims 7-13 and 16 are rejected under 35 U.S.C. § 101 as allegedly being directed to non-statutory subject matter. Claims 4, 6, 10, 11, and 12 remain rejected under 35 U.S.C. § 102(e) as being anticipated by Baentsch et al (U.S. Patent Application Publication No. 2002/0059475, hereafter "Baentsch"). Claim 7 is rejected under 35 U.S.C. § 102(b) as being anticipated by Kwong et al (U.S. Patent No. 6,289,506). Claims 1-3, 14, 15, and 17 are rejected under 35 U.S.C. § 103(a) as being unpatentable over Zee (U.S. Patent Application Publication No. 2003/0005425) in view of Baentsch. Claims 5 and 13 are rejected under 35 U.S.C. § 103(a) as being unpatentable over Baentsch in view of Kwong. Claims 8, 9, and 16 are rejected under 35 U.S.C. § 103(a) as being unpatentable over Kwong in view of Baentsch.

As a preliminary matter, Applicant thanks the Examiner for removing the 35 U.S.C. § 101 rejections of claims 1, 2, 3, 4-6, 14, 15, and 17 and the objections to claims 5, 6, 9, and 10 made in the Office Action dated October 19, 2006.

**I. Rejections under 35 U.S.C. § 101**

In the previous Office Action dated October 19, 2006, the Examiner asserted that claims 7, 10 and 12 are directed to non-statutory subject matter, and fail to produce a useful, concrete and tangible result.

In the Amendment filed on January 19, 2007, Applicant amended claims 7, 10, and 12 in order to more clearly define the invention.

In response, the Examiner asserts that:

[T]he claimed subject matter does not produce a tangible result because the claimed subject matter fails to produce a result that is limited to having real world value rather than a result that may be interpreted to be abstract in nature as, for example, a thought, a computation, or manipulated data...The act of execution itself does not produce a result since there can be an error during execution.

Applicant respectfully disagrees with the Examiner's position, and respectfully submits that the claims are statutory in nature, and that the Examiner's § 101 rejection is erroneous.

MPEP 2106.IV.C(2) details the review that should be made to determine whether a claimed invention falls within 35 U.S.C. § 101 judicial exceptions, since claims directed to nothing more than abstract ideas (such as mathematical algorithms), natural phenomena, and laws of nature are not eligible for patent protection. MPEP 2106.IV.C further states that while abstract ideas, natural phenomena, and laws of nature are not eligible for patenting, methods and products employing abstract ideas, natural phenomena, and laws of nature to perform a real-world function may well be.

MPEP 2106.IV.C.2 then outlines the various evaluations that must be made to determine whether a claimed invention covers either a 35 U.S.C. § 101 judicial exception (abstract ideas, natural phenomena, and laws of nature) or a practical application of a 35 U.S.C. § 101 judicial exception. One of these evaluations includes determining whether the practical application produces a useful, concrete and tangible result, wherein "the tangible requirement does not

necessarily mean that a claim must either be tied to a particular machine or apparatus or must operate to change articles or materials to a different state or thing. However, the tangible requirement does require that the claim must recite more than a 35 U.S.C. § 101 judicial exception, in that the process claim must set forth a practical application of that judicial exception to produce a real-world result."<sup>1</sup>

The instant application does not pertain to an abstract idea (such as mathematical algorithms), a natural phenomenon, or laws of nature.

It is well established that in order to satisfy the requirements of 35 U.S.C. § 101, a claimed invention must produce a "useful, concrete and tangible result." *State Street*, 149 F.3d at 1373, 47 USPQ2d at 1601-02. In the instant application, a method for improving the performance of a Java platform is presented.

In independent claim 7, the performance of the Java platform is improved by precompiling a class file included in a standard class library into an extended class library file including a machine instruction; the extended class library file executing the machine instruction and executing a Java application file by using at least one of a Just-In-Time (JIT) compiling method and an interpreting method. The improvement in performance of the Java platform would be the "tangible result" of the invention as claimed in independent claim 7.

Similarly with respect to independent claim 10 and analogous independent claim 12, the improvement in performance of the Java platform is achieved by converting a Java file or a Java source file into a machine instruction including an operand in which symbolic reference information is inserted in place of an address, the improvement in performance of the Java

---

<sup>1</sup> MPEP 2106.IV.C(2)(b).

platform being the “tangible result” of the invention. Accordingly, Applicant respectfully submits that the 35 U.S.C. § 101 rejection is clearly erroneous, and respectfully request that the Examiner remove this rejection.

## II. Prior Art Rejections

### A. Rejection of claims 4, 10, and 12

In the previous Office Action dated October 19, 2006, the Examiner asserted that Baentsch discloses “wherein a symbolic reference information indicates a specific class, field or method of an object, and method information of the method comprises an attribute of a code formed of the machine instruction having an operand in which the symbolic reference information is inserted in place of an address”, and cited paragraphs [0008]-[0009] of Baentsch as allegedly disclosing this feature of independent claim 4 and analogous independent claims 10 and 12.

In the Amendment filed on January 19, 2007, Applicant submitted that there is no disclosure in Baentsch of a machine instruction having an operand in which the symbolic reference information is inserted in place of an address, and that contrary to the Examiner’s assertion, the cited portion of Baentsch discloses that during a downloading and linking process, the interpreter of the Java Virtual Machine (JVM) looks up target items referred to in the bytecode instructions by name and replaces the references with corresponding addresses (paragraph [0008], lines 11-15), as opposed to the claimed invention which inserts a symbolic reference in place of an address.

In response, the Examiner now cites paragraph [0023] of Baentsch as allegedly disclosing this feature of the claim and asserts that:

[S]ince the actual address is not known, a reference is inserted in the place of an address. During the linking process is when the referenced items are then replaced by the run-time specific identifiers.<sup>2</sup>

Applicant respectfully disagrees with the Examiner's position, and respectfully submits that the claims are not anticipated by Baentsch.

Paragraph [0023], as best understood, merely discloses that the code section 20 references all symbols [whose] actual addresses are not known before linking time. Nowhere does this cited portion (or any other portion) of Baentsch teach or suggest that a machine instruction having an operand in which the symbolic reference information is inserted in place of an address as recited in independent claim 4 and analogously recited in independent claims 10 and 12.

Independent claim 4 has been amended to read "A Java class file in a class library which is executed on a Java platform". A characteristic of the instant invention is to incorporate an attribute of a code formed of the machine instruction into a Java class file structure in a Java class library. Paragraph [0023] of Baentsch describes that JavaCard cap files 10 comprise code section 20 and the code section 20 bytecode instructions. In paragraph [0072] of the instant invention, an attribute of code of a standard class file specification includes machine code information instead of information on Java bytecodes. JavaCard cap files 10 cited by the Examiner, are different from the Java class file structure. The difference between the Java bytecode and the machine code is described in paragraph [0025] of the instant application.

---

<sup>2</sup> Page 14 of the Office Action dated April 3, 2007.

**B. Rejection of claim 7**

In the previous Office Action, the Examiner asserted that Kwong discloses “precompiling a class file included in a standard class library into an extended class library file including a machine instruction”, and cited column 5, lines 8-67 of Kwong as allegedly disclosing this feature of the claim.

In the Amendment, Applicant submitted that there is simply no disclosure in Kwong of precompiling a class file included in a standard class library into an extended class library file including a machine instruction as required by claim 7.

In response, the Examiner now cites column 6, lines 29-31 as allegedly disclosing this feature of the claim and asserts that:

The base classes are considered to represent the standard library and the extension classes are considered to represent the extended library.

Applicant respectfully disagrees with the Examiner’s position, and respectfully submits that there is no disclosure in Kwong of precompiling a class file included in a standard class library into an extended class library file including a machine instruction as required by claim 7. Although Kwong may disclose elements in a computer system equipped to interpret and compile class files, there is no teaching or suggestion in Kwong of precompiling a class file included in a standard class library into an extended class library file including a machine instruction as required by claim 7.

Additionally, Java Base Classes 206 and Java Standard Extension Classes 210 in Kwong are described to be the implementation of its respective API. Column 5, lines 27-33 of Kwong states that Java Base API 204 provides the basis language, utility, network etc., and Java

Standard Extension API 208 extends the capability of Java beyond the Java Base API. As such, in Kwong, Java Standard Extension Classes 210 merely extend the function of Java Base Classes 206, but does not precompile the class file included in the standard class library.

**C. Rejection of claim 1**

In the previous Office Action, the Examiner asserted that Zee discloses all of the features of claim 1 except for a Java Virtual Machine (JVM), and asserted that it would have been obvious to one of ordinary skill in the art to use the JVM disclosed by Baentsch as a software-only platform running on top of the hardware-based platform of processing systems 30 and 32 disclosed by Zee.

In the Amendment, Applicant submitted that, contrary to the Examiner's assertion, there is no teaching or suggestion in Zee of an extended class library which includes a class file of a machine code obtained by precompiling a class file included in a standard class library as recited in claim 1.

In response, the Examiner asserts that:

The designated class file is considered to represent to represent a file from the standard library. Zee then discloses that "The compile-on-demand server 20 or 22 selects an AOT compiler which can produce a native component for the data processing system..." This is considered to represent the step of precompiling.

Applicant respectfully disagrees with the Examiner's position, and submits that there is no teaching or suggestion in Zee of "an extended class library which includes a class file of a machine code obtained by precompiling a class file included in a standard class library" as recited in claim 1.

Zee discloses compiling a downloaded Java class file into a native component and saving the native component. There is no teaching or suggestion in Zee of precompiling a class file included in a standard class library as required by independent claim 1. Further, Zee does not disclose a class library comprising machine code as claimed.

Accordingly, respectfully submits that independent claims 1, 4, 7, 10, and 12 as well as dependent claims 2, 3, 5, 6, 8, 9, 11, and 13-17 should be allowable because the cited references do not teach or suggest all of the features of the claims.

In view of the above, reconsideration and allowance of this application are now believed to be in order, and such actions are hereby solicited. If any points remain in issue which the Examiner feels may be best resolved through a personal or telephone interview, the Examiner is kindly requested to contact the undersigned at the telephone number listed below.

The USPTO is directed and authorized to charge all required fees, except for the Issue Fee and the Publication Fee, to Deposit Account No. 19-4880. Please also credit any overpayments to said Deposit Account.

Respectfully submitted,



Mark E. Wallerson  
Registration No. 59,043

SUGHRUE MION, PLLC  
Telephone: (202) 293-7060  
Facsimile: (202) 293-7860

WASHINGTON OFFICE  
23373  
CUSTOMER NUMBER

Date: June 4, 2007